USB2.0 综合采集控制板 RBH8281

250Ksps 光电隔离
16 位 16 通道 AD
4 通道 DA 输出
4 通道光电隔离脉冲输入
16 通道数字输入 16 通道数字输出
USB 接口板 RBH8281

USB 接口嵌入式 DSP 实时控制板

硬件使用说明书

北京瑞博华控制技术有限公司

二 00 八年十二月

250Ksps 光电隔离

16 位 16 通道 AD

4 通道 DA 输出

4 通道光电隔离脉冲输入

16 通道数字输入 16 通道数字输出

USB 接口板 RBH8281

硬件使用说明书

一、性能特点：

本板特别适合于 Windows 下实时控制。

在 Windows 下一方面实现良好的用户界面和数据管理功能，另一方面又要实现实时控制，并且实现用户自己软件编程，RBH8281 能非常完美地解决这个问题。

本板采用上下位机的方式工作，板卡上带 FreeScale 的 DSP 芯片，该芯片为 DSP56F803，主频为 80MHz，带大容量的 FLASH ROM 和 RAM，并且带多通道的 AD、DA、开关量输入、开关量输出以及脉冲量输入，这些资源都受 DSP 控制，用户通过对 DSP 的编程，就可以实现实时信息采集和处理功能。

板上还带有高速的 USB2.0 接口，与 PC 机之间实现高达 480MHz 的通讯速率，用户可以通过编写 Windows 程序和 DSP 程序，实现两者高速、双向的数据通信功能，因此可以实现软件界面友好、实时控制功能强的特点。

本板提供成熟的硬件技术支持。板卡上带有高密度的 CPLD 芯片，可以根据用户的需要定制各种特殊的功能，并且为用户调试好各种功能。

本板提供成熟的软件技术支持。无论是 Windows 下的接口技术、硬件的驱动程序、USB 接口的底层驱动程序、还是下位机的 DSP 软件、CPLD 应用软件等，本公司全部封装好，最后提供非常简单的上位机 Windows 下的 DLL 应用程序和下位机的 C 语言程序，并提供上位机的应用例程和下位机的应用例程，包括下位机对全部硬件操作的例程也完全提供，因此，用户只需基本的 Windows 编程技术基础和 C 语言基础，就可以完成 Windows 下的实时控制功能。

本板通过采用高速高精度 AD 芯片、高精度的放大器、高密度 FPGA 逻辑芯片、精细地布线以及优良的制版工艺，实现了高速、高精度实时数据采集，具有以下性能特点：

1、 完备的硬件和软件支持，保证在 Windows 下的实时控制功能。

2、 下位机用 C 语言开发，提供全部的硬件接口源程序和实时控制例程，提供成熟的下位机编程环境，为用户对 DSP 开发提供全面的技术支持。用户不需要对 DSP 开发、USB 接口开发、AD,DA,脉冲测量、开关量输入检测、开关量输出控制等进行深入了解就可以进行自己任务的开发。

3、 上位机开发可以直接在 VB,VC 等常用的环境下进行，不需要了解 USB 接口、驱动程序、通讯协议等，本采集卡提供完备的软件接口和非常容易的编程方法，可以实现在 Windows 下友好的软件编程和与 DSP 得高速双向通信功能。

4、 采用 USB2.0 接口,光电隔离模拟量采集。一方面满足高速采集的速率要求，另

一方面满足USB接口在工业现场的可靠性要求。

5、 灵活的供电电源方式，能够满足不同现场的需要。用户可以提供外接的模拟电源，这样就可以实现模拟信号与计算机系统之间的隔离电压高达2500V以上，大大提高系统的可靠性。

6、 AD幅值采集高精度：RBH8281控制卡为16位采集精度,差分非线性误差小于±2LSB;相对精度优于0.006％。

7、 AD高速度： 实际总的采集贯通最大速率可以达到250KSPS,最低速率可为1Hz 。采集方式为连续采集，为用户实现各种触发与启动方式提供条件。用户采集的数据量仅仅决定于用户的硬盘，为用户海量存储提供条件。

8、 AD采集定时高精度：本板直接在CPLD控制下工作，由硬件时钟直接控制采集与传输，采集精度与晶振精度相同，缺省定时精度误差小于50PPM。对于有特殊要求的用户，可以通过更换晶振的方式，达到0.1PPM精度，甚至更高精度。

9、 软件校准：将校准信息存储在板卡上，用户不用打开仪器设备就可以进行校准，使用方便。

10、 丰富的备用扩展资源：板上CPLD资源非常丰富，可以为用户的特殊需求进行定制，如旋转编码器接口、脉冲周期测量接口、PWM输出接口、外同步接口、触发记录接口、开关量控制接口等。

二、功能与指标

AD的性能指标：

● AD采样精度：RBH8281控制卡为16位采集精度,差分非线性误差小于±2LSB;相对精度优于0.006％。

● AD通道数：单端方式16通道。

● AD系统数据采集实际贯通率：250KSPS(250000 SAMPLE PER SECOND)。

● 模拟采集的定时精度：缺省情况下为50PPM,特殊要求可以定制

● AD输入电压范围：-5V到+5V或0－10V。可以按照用户要求定制

● AD输入阻抗：1000千欧

● 模拟输入安全电压：±25伏。当超过AD输入超过信号量程时，不会引起硬件损坏，但会导致削顶现象。

● 抗静电电压：2000伏

● 采集方式：连续采集

● 隔离电压：隔离电压为2500伏。

● 模拟部分隔离方法：数字隔离方式。在模拟端，A/D芯片将模拟量信号变成数字量后通过光藕送到计算机内，隔离方式采用数字隔离模式，大大提高采集精度。

● 隔离端外接电源：当用户采用外接电源方式时，用户输入的电源电压是－15伏和＋15伏。

DA的性能指标：

● DA采样精度：12位

● DA通道数：4通道。

- DA 输出的编码方式：偏移码，0－4095，0 输出 DA 最小值，4095 输出最大值。
- DA 输出的时间延迟：小于 10US。
- DA 输出电压：－5V 到＋5V，或 0－10V,也可以用户自由调整。
- 上电输出电压模式：中间输出或最小输出，用户购买时指定。当采用中间输出时，上电输出 DA 的值为 2048，当采用最小值输出时，上电输出的 DA 值为 0。
- 隔离电压：隔离电压为 2500 伏。
- 模拟部分隔离方法：数字隔离方式。
- AD 部分与 DA 部分共用相同的电源，两者之间没有电气隔离。

开关量指标:

- 16 路数字量输入。
- 开关量输入，TTL 电平。
- 16 路数字量输出,上电复位清零功能 。
- 开关量输出：TTL 电平。
- 光电隔离方式，隔离电压 2500V

脉冲指标:

- 通道数：4。
- 隔离方式：光电隔离。
- 脉冲速度：第一通道的速度为 10MHz，第 2、3、4 通道的速度为 40KHz。

电源：

- USB 接口提供 USB 接口部分的电源。
- 模拟部分外接±15V 电源。
- 开关量输入采用光电隔离模式，信号电流为输入 5-10mA
- 开关量输出采用光电隔离模式，外接电源+5V
- 脉冲信号输入采用光电隔离模式，信号输入 5-10mA。

工作环境

- 工作温度：0－70℃
- 环境湿度：95％以内

形状尺寸

- 外形尺寸：250mm×125mm×18mm
- 定位孔位置：240mm×115mm
- 定位孔尺寸：4×Φ3.5mm

三、AD 板工作原理简介

RBH8281 板的硬件组成原理框图如图 1 所示。

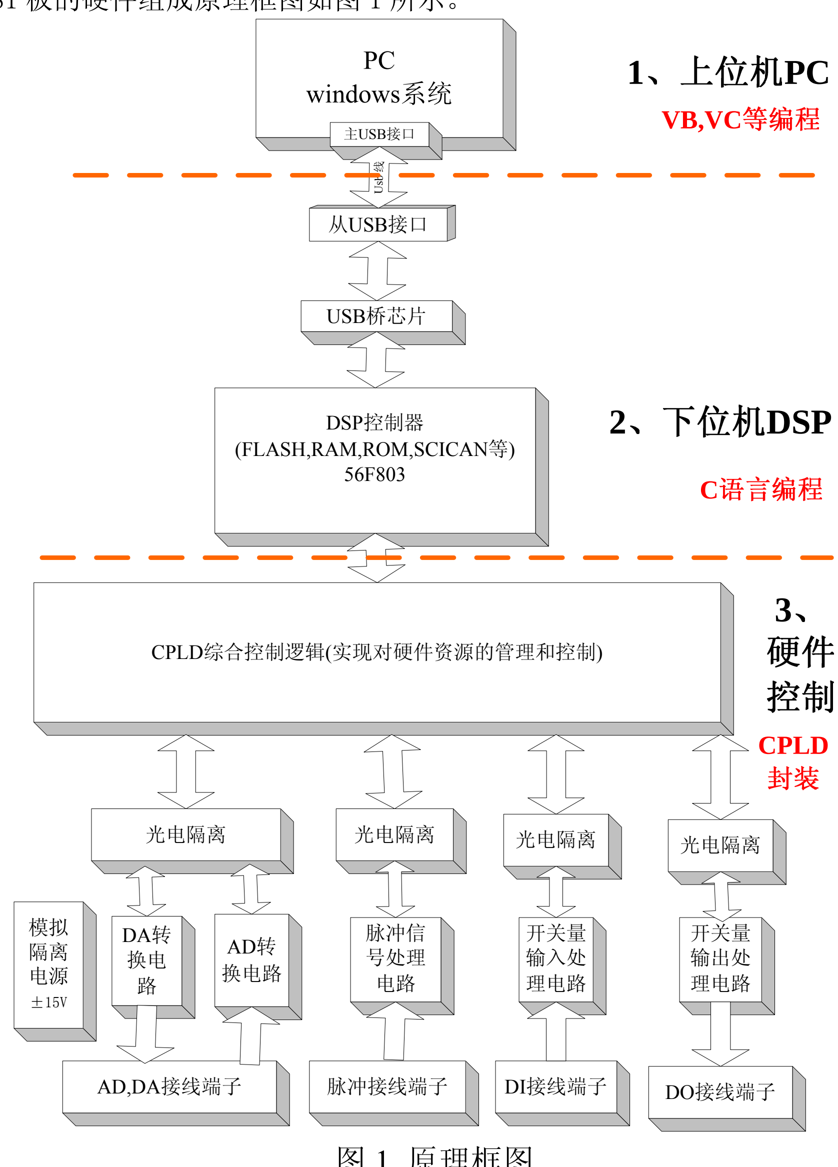


图 1 原理框图

从图 1 可以看出，系统分为 3 部分，第一部分是上位机，也就是 PC 机，包括驱动程序、DLL、应用程序接口、主 USB 接口等，用户在 Windows 下编写自己的应用程序；第二部分是下位机，也就是 DSP 系统，用户用 C 语言编写实时数据采集和控制程序，通过一根 USB2.0 的连接电缆线与第一部分相连；第三部分是 CPLD 管理的硬件资源，包括 AD,DA, 开关量输入和开关量输出、脉冲信号采集等，这部分不需要用户进行二次开发，本公司提供全部的 CPLD 控制，并进行封装，提供 DSP 的接口软件子程序，用户只需要调用相应的子程序就可以实现对全部硬件资源的控制。第二部分的 DSP 与第三部分的 CPLD 通过内部并行数据总线相连，通过 DSP 实现对 CPLD 的通讯和控制，用户通过软件就可以访问外部的硬件资源，实现实时采集和控制功能。

第三部分的 CPLD 与硬件资源之间全部通过光电隔离的方式连接，而且个各个部分独立隔离，从而保证本卡在工业现场的可靠性，隔离电压达到 2500V。

系统的工作过程是：上电后，DSP 程序就按照用户编写的应用程序开始运行，一方面

采集各种硬件资源的信息，另一方等待上位机通过 USB 接口传送的命令和数据；当用户运行上位机 Windows 应用程序后，应用程序就可以调用本公司提供的接口程序向下位机发送命令和数据，该命令和数据就会通过 USB 电缆线自动传送到 DSP 得 C 语言程序中；下位机接收到上位机的命令和数据后就开始按照要求进行采集和控制，并且可以高速、定时地将采集数据、控制信息等数据源源不断地通过 USB 接口送到上位机，同时仍然在检测上位机发送给下位机的命令和数据；上位机在接收到下位机的实时数据后就可以进行信号检测、状态显示等功能，完成 Windows 应用程序的各种功能。

由于 DSP 的主频高达 80MHz，USB2.0 的速度高达 480Mpbs，因此，能够满足绝大多数的工业控制的要求。

四、硬件使用方法

1、操作元件布置

本板的操作元件布置如图 2 所示。为了便于用户使用，在板卡上已经有详细的标识。

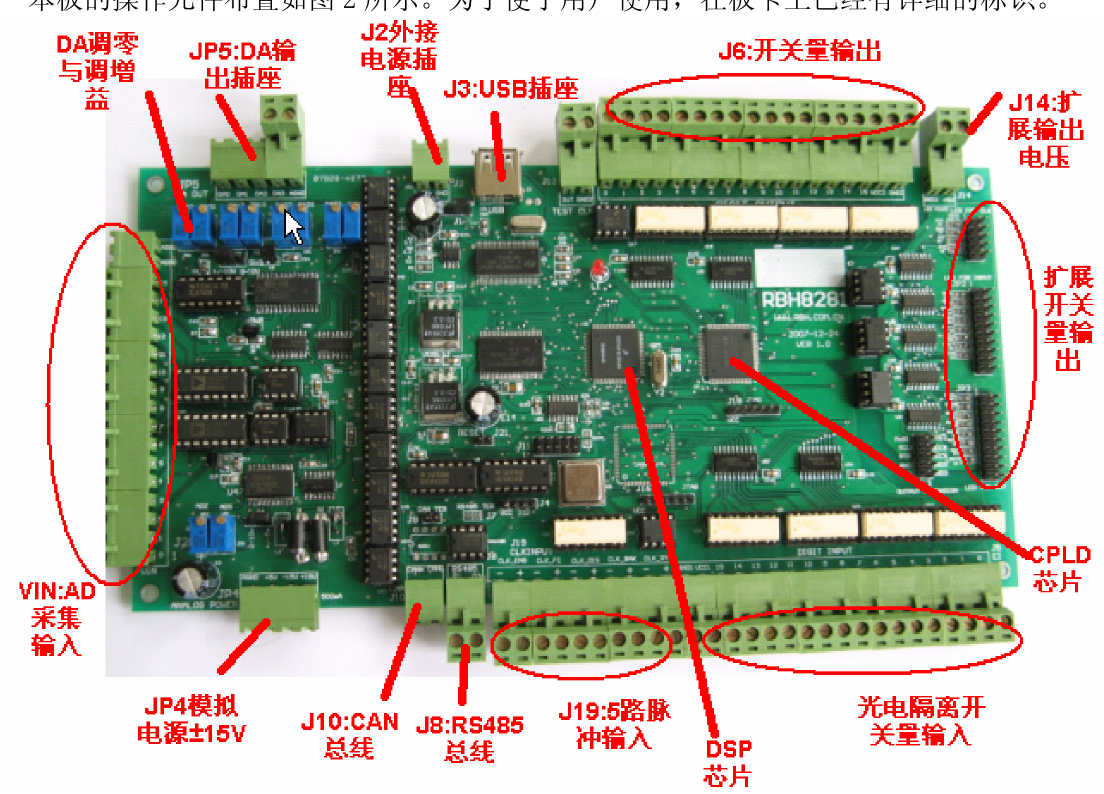


图 2 实物及接头定义

各个接头的定义是：

模拟量信号：图 2 的左侧

最左边的 VIN 是 18 个接线端子包括 16 个通道的 AD 输入和 2 个模拟地线 AGND。图中标注 0-15 对于 1-16 通道。模拟量输入的调零与调增益电位器在途中的左下侧。

左下 JP4 是模拟量电源输入，输入±15V 的电源，建议尽可能采用线性电源，会提高采集精度。

左上 JP5 是模拟量 DA 输出端子。

模拟量采集与模拟量输出通过中间的光耦实现光电隔离。

总线扩展：CAN 总线和 RS485 总线

如图 2 的中部下端所示，J10 是 CAN 总线输出端子，J8 是 RS485 的输出端子。这两个通讯信号是 DSP56F803 芯片提供的通讯功能，便于用户进行联网控制。

脉冲量输入：

图 2 中下部所示，有 5 个脉冲量虽然，一路高速输入，速度到达 10MHz，4 路低速输入，最高速度为 40KHz。脉冲信号的输入是各个通道独立隔离，分别正向和负向。

开关量输入：

图 2 右下部所示，信号从 J5 输入。

开关量输出：

图 2 右上所示，信号从 J6 输出，开关量输出的电源也在 J6 上输入。

扩展开关量输出：

图 2 的右侧是开展的开关量输出功能，可以实现 SPI 总线的输出功能，还有 40 路光电隔离的开关量输出，便于用户启动信号灯等控制。

板卡的尺寸定义如图 3 所示。

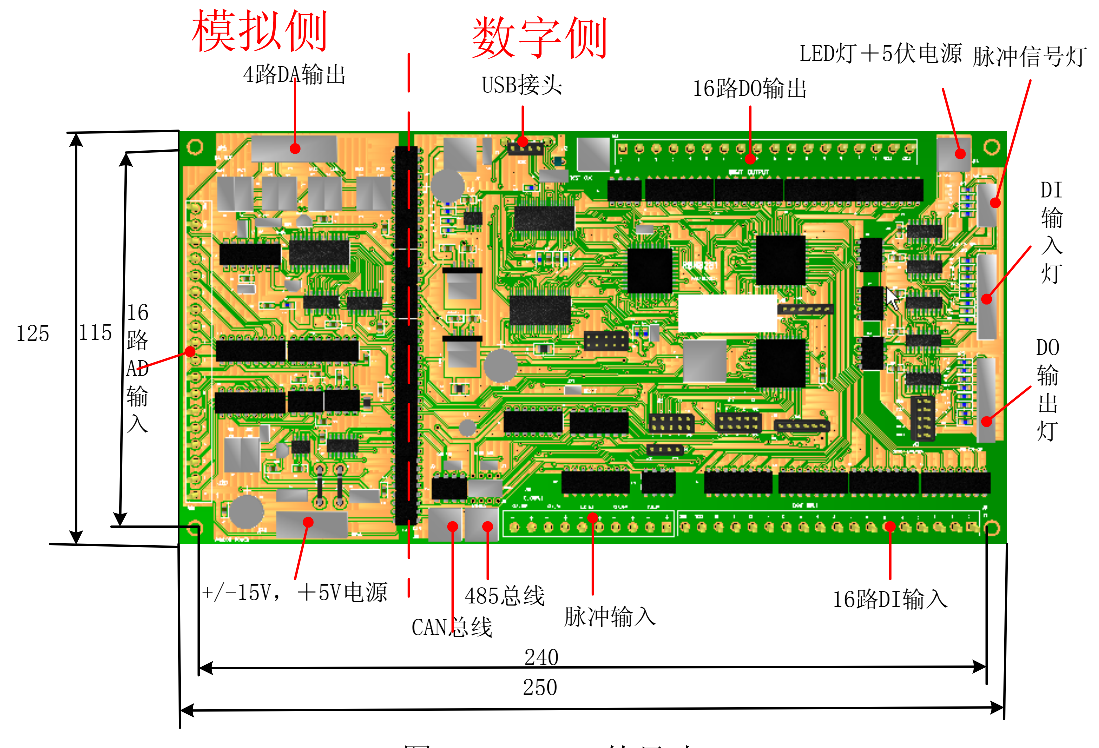


图 3 RBH8281 的尺寸

2、16 通道开关量输入的使用说明

开关量光隔离输入共阳极输入方法,外接隔离电源 EVCC1，信号从 EI0-EI15 输入，在限流电阻作用下，通过电流检测的方式，实现开关量检测功能。内部限流电阻为 1K 欧。一般 TTL 电平输入，建议 EVCC1 为＋5V，对应其它电平，要求导通时的电流为 5－15mA。用户可以通过接不同的外接电源 EVCC1 和外部串不同电阻的方法，实现对不同开关量信号的检测功能。当有电流流过时，内部检测到 1，否则检测到 0。

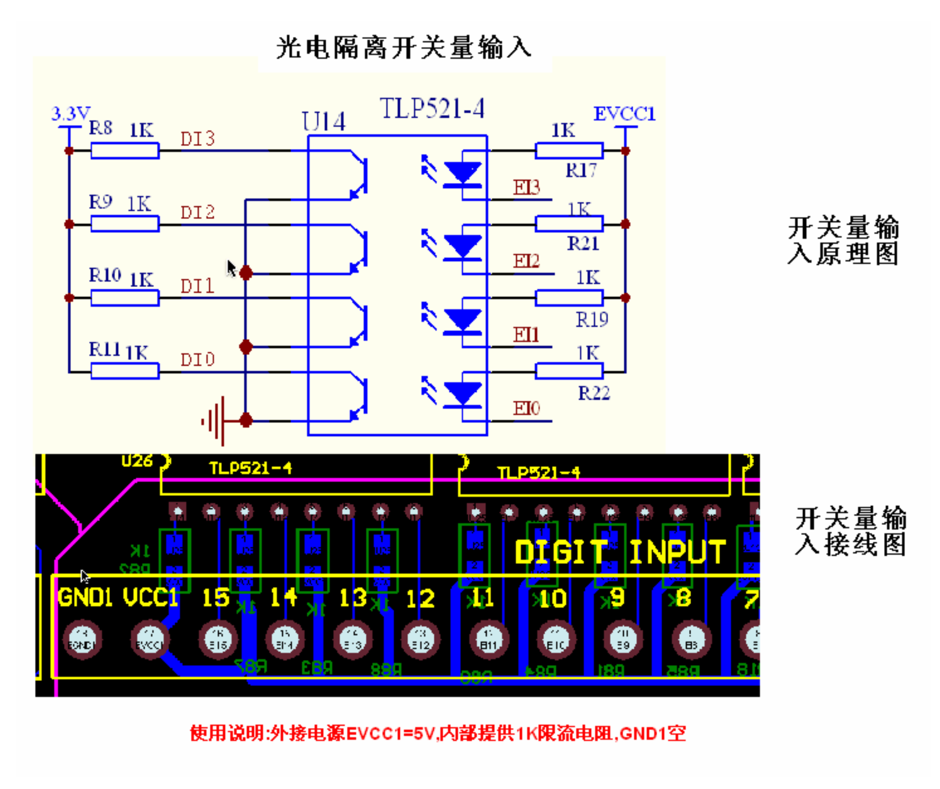


图 4　16 通道开关量输出的使用说明

3、16 通道开关量输出的使用说明

开关量光隔离输出采用共地的方法输出。内部控制信号 DO0 为低电平时，输入光耦导通，输出 EO0 与地线 EGND2 导通，输出低电平；当 DO0 为高时，EO0 与 EGND2 断开，在微弱上拉电阻 47K 的作用下与电平 EVCC2 相同，输出高电平。由于采用微弱上来，可以把输出当集电极开路来使用。

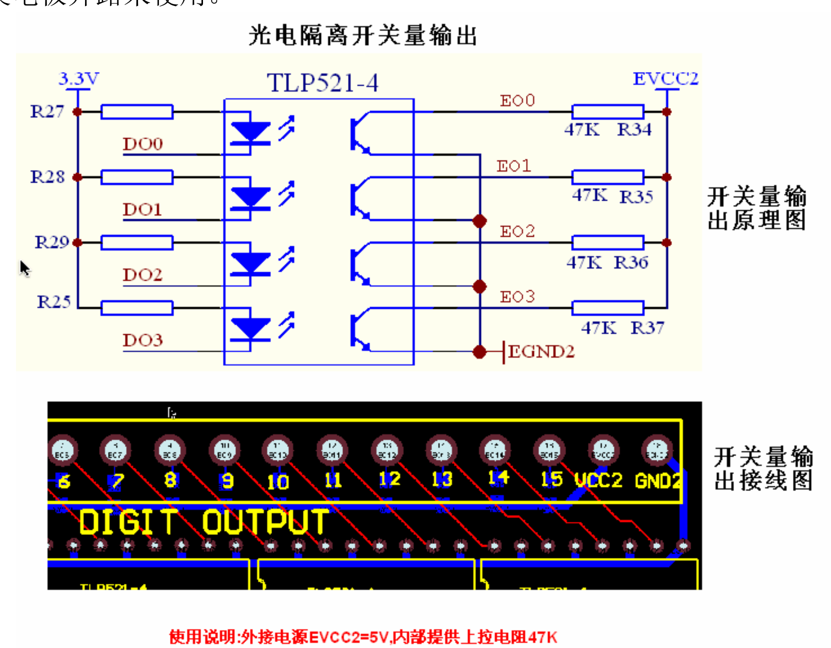


图 5 光电隔离开关量输出

4、4 通道脉冲输入的使用说明

4 通道的脉冲量采用独立隔离的方式输入，其中 DY 输入为高速脉冲输入，输入的频率范围达到 10MHz，ENG,FC,DIS 为低速脉冲输入输入频率范围小于 25kHz。 内部的限流电阻为 510 欧。

光电隔离脉冲量输入

脉冲量输入原理图

脉冲量接线图

使用说明:+接信号的正端,-接信号的负端,要求内部限流电阻510欧,输入电流5-15mA,用户可以外接限流电阻以匹配各种输入脉冲信号.

图 6 脉冲量输入电路

5、40 通道光电隔离 LED 灯控制输出使用说明

40 通道光电隔离 LED 灯的输出通过统一光隔的方式和 SPI 技术实现的。输出需要外部电源从 J14 输入，电源电压为＋5 伏。输出的第一个 8 通道为 JP1，第二个 8 通道与第三个 8 通道从 JP2 输出，第四个 8 通道与第五个 8 通道从 JP3 输出。

信号输出通过串接 1K 电阻到接线端子上，因此可以直接接 LED 信号灯。为了便于使用，在每个接线针上对接有电源线与地线，其中 EVCC3 就是＋5V 电源，EGND3 就是地线。用户可以多种方式实现对信号灯的控制。

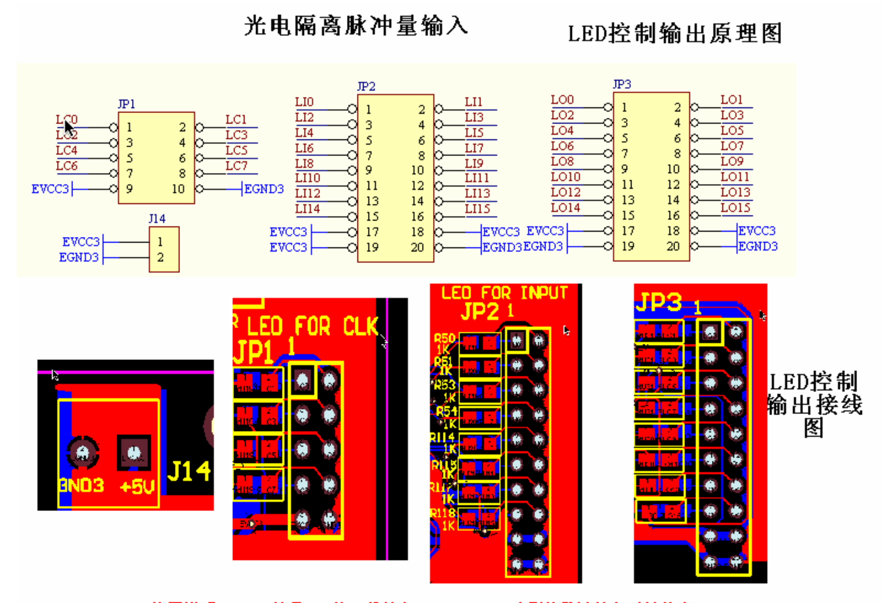


图 7 40 通道 LED 开关量控制输出

6、光电隔离模拟量采集输入与模拟量输出的使用说明

模拟量与 CPU 系统完全隔离。模拟量系统的工作需要外接电源＋15V，－15V，＋5V 这三个，其中 AGND 为模拟电路的地线，该地线与模拟量的电源的地线、模拟量输入地线、模拟量输出地线相通。



图 8 模拟量外接电源

模拟量输出从 JP5 输出，其中 AGND 为地线、DA0 为第一通道 DA 输出，DA1 为第二路 DA 输出，DA2 为第三路 DA 输出，DA3 为第四路 DA 输出。

DA0Z,DA0S:为第一通道 DA 输出的零点与增益调整电位器。

DA1Z,DA1S:为第二通道 DA 输出的零点与增益调整电位器。

DA2Z,DA2S:为第三通道 DA 输出的零点与增益调整电位器。

DA3Z,DA2S:为第四通道 DA 输出的零点与增益调整电位器。

模拟量输入从 VIN 输入，其中 AGND 为地线，1－16 为 16 通道的模拟量输入。信号输入范围是－5 伏到＋5 伏。信号输入采用单端方式。

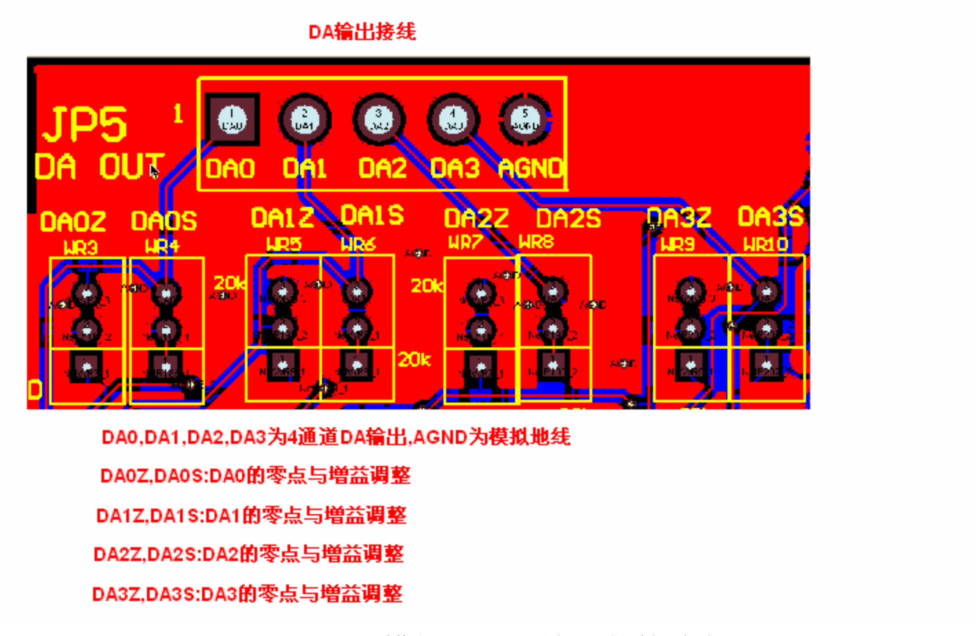


图 9 模拟量 DA 输出的接线与调整

五、软件使用说明

USB 总线有即插即用的特点，为用户使用本卡提供了很多方便，对于大多少用户，可以直接采用本公司提供的驱动软件，可以实现数据采集功能。

本板提供了很完善的 WIN98/2000/NT/XP 驱动程序，采用动态链接库的方式，用户使用方便、快捷，所提供的 DEMO 软件，能满足大量的实际需要，如实时控制、波形显示、波形记录等。

在 Windows 下编程,有两种编程方式,一种是采用查询方式,可以实时读取当前信号的幅值,以及开关量状态,这种方式特别适合于工业现场的实时控制;另一种方式是采用硬件定时采集的方式,通过调用本公司提供的动态连接库,可以实现在 Windows 下高速、实时、连续采集信号。具体编程的接口函数请参考光盘上编程指南相应的文档。

1、 上位机演示例的说明

为了便于用户编程方便，本板提供编程例程。

例程软件用VB编写，光盘中有详尽的源代码与说明。

通过调用DLL，例程演示了如何启动采集，如何高速连续采集并保存数据。

对开关量的的采集与显示，已经数据从数据包分离的方法也有详尽的介绍。

图 10 VB测试程序界面

图10是VB测试程序的界面，可以通过VB程序实现对下位机硬件的操作，如开关量输出、LED灯控制、DA输出测试等，对于下位机AD采集、脉冲信号采集、开关量输入采集与本公司原来的采集系统结构完全相同，具有高速连续采集的特点。

下面是开关量输出的例程源程序

```
Private Sub cmdSW0_Click() '开关量输出例程
    Dim ilong  As Long
    InBuff(0) = 12 '通用功能
    InBuff(1) = 3 '命令号
    InBuff(2) = Val(txtswChn.Text) '通道号
    If (InBuff(2) > 4) Then InBuff(2) = 4
    InBuff(3) = Val("&H" + txtSwValue.Text) '值
    ilong = DllIOCtl(100, InBuff(0), 100, OutBuff(0))
End Sub
```

下面是LED灯控制输出的源程序

```
Private Sub cmdLedOutput_Click() 'LED灯控制输出
Dim ilong  As Long
Dim i As Integer
    InBuff(0) = 12 '通用功能
    InBuff(1) = 5 '命令号
    InBuff(2) = Val(txtLedChn.Text)
    If (InBuff(2) > 4) Then InBuff(2) = 4
    InBuff(3) = Val("&H" + txtLedValue.Text)
    ilong = DllIOCtl(100, InBuff(0), 100, OutBuff(0))
End Sub
```

下面是DA输出的源程序

```
Private Sub cmdDAOutput_Click() 'DA输出
```

```
Dim ilong  As Long
Dim i As Integer
    InBuff(0) = 12 '通用功能
    InBuff(1) = 4 '命令号
    InBuff(2) = Val(txtDAChn.Text)
    If (InBuff(2) > 3) Then InBuff(2) = 3
    ilong = Val("&H" + txtDAValue.Text)
    ilong = ilong And &HFFF
    InBuff(3) = ilong And 255
    InBuff(4) = Int(ilong / 256) And 255
    ilong = DllIOCtl(100, InBuff(0), 100, OutBuff(0))
End Sub
```

下面是启动采集的源程序

```
Private Sub cmdStart_Click()
Dim i
    NumBuf = 10 '缓冲区个数
    NumSamp = 500 '每个缓冲区采样点数,
    BegChn = 0 '起始通道
    NumChn = 16 ''通道数
    FrqSamp = 10000 '采样频率
    FrqFilter = 0 '滤波器频率
    AmpGain = 1 '放大器增益
    ReDim ADBuf(NumSamp * NumChn + 1) As Integer '每个采样块的大小,该数组在
后面的取数据程序中要使用
    ReDim RecordBuf(RecordBlock * (NumSamp * NumChn)) As Integer  '内存保存
数据
    i = DllStartIntr(NumBuf, NumSamp, BegChn, NumChn, FrqSamp, FrqFilter,
AmpGain)
    TimerAD.Enabled = True '启动电压监视
     For i = 1 To NumSamp * NumChn + 1
        ADBuf(i) = 0
    Next i
End Sub
```

2、 下位机演示例的说明

```
int main()
{
    SysInit();//系统初始化
    archEnableInt();
    while(true){
       Poll();
    }
}

void Poll(void)
{     copReload();//看门狗功能
      STA = *pDI;
      //处理来自于USB的信息
      if((STA&4)){                     //FlagC = EP2 PF,USB-
         FillEP6Flag=false;            //禁止向EP6填充数据
         SelectEP2();
         for(PollI=0;PollI<32;PollI++) { //64 bytes,读取数
            EP2[PollI] = *pUSB;          //读取EP2的64个字
         }
         SelectEP6();//选取EP6
```

图11 下位机软件开发界面

如图11所示，下位机软件在CodeWarrior环境下进行，采用简单的C语言，在该环境下可以实现下位机软件的编辑、编译和下载到板卡上。

如图11所示，下位机软件的主程序主要包括两部分内容，一个是进行系统初始化，另一个就是不断地查询上位机状态，一旦检测到上位机命令，就进行相应的操作。实时信息采集和处理在一个硬件定时器中断中进行。硬件定时器中断如下所示：

```
//-----由CPLD产生的硬件定时器中断-------------------
 void FDownButtonAISR (void)
{     temp = *pDI;
    if(Started && FillEP6Flag && ( ! (temp&1) || ! (temp&2) ) ) {
   //---------A/D转换程序开始
if(IdFirst<=0){
ADR[ChnCounter]=ReadADResult();//读取采集结果
ChnCounter++; if(ChnCounter==ADNumChn)ChnCounter=0;//下一个通道
}else IdFirst--;
StartAD();//启动A/D转换
if(IdFirst>0) IntI=3-IdFirst;//下一个物理通道
   else          IntI=(ChnCounter+2)%ADNumChn;
   SetChn(IntI);//转换到相应通道
   //---------A/D转换程序结束
   ReadClk();//读取脉冲数据,
//下面将采集的结果送到PC机
```

```
    if((IdFirst<=0)&&(ChnCounter==0)){
        IntI=*pCSDI0;//采集开关量输入的低8位
        IntJ=*pCSDI1;//采集开关量输入的高8位
        IntJ=((IntJ & 0xff)<<8)+(IntI &0xff);//合成16位开关量输入
        *pUSB=IntJ;     //第一个通道采集开关量,送到上位机
        IntI=(temp>>3)&0x1f;//FLAG10MS,CLKDY,CLKFC,CLKENG,CLKDIS
        *pUSB=IntI; //第二通道存放,低4位存放当前脉冲量的当前状态
    //FLAG10MS是BIT.4,表示10MS的时间是否到,也表示采集DY的数据是否完成
if((IntI&0x10)==0x10){//DY测量完成,FLAG10MS=1表示采集DY数据完成,10MS时间到
      IntI=ReadCpld(11);
      IntJ=ReadCpld(12);
      SpeedDY=IntI+(IntJ<<8);//脉冲计数值
      IntI=ReadCpld(6);//读取10毫秒内完整周期的周期和的低8位
         IntJ=ReadCpld(7);//读取10毫秒内完整周期的周期和的高8位
         PeriodDY=(IntJ<<8)+IntI;//完整周期的周期和
         *(pBase0+10)=0;//启动下一次DY检测
     }
        for(IntI=2;IntI<10;IntI++){      //可以根据需要灵活配置
           if(IntI==2)*pUSB=SpeedEng;  //第三通道存放CLKENG的转速
             if(IntI==3)*pUSB=CounterFC; //第四通道存放CLKFC的转速
             if(IntI==4)*pUSB=CounterDis;//第五通道存放CLKDIS的脉冲数
             if(IntI==5)*pUSB=SpeedDY;   //第通道存放CLKDY的脉冲数
             if(IntI==6)*pUSB=PeriodDY;
             if(IntI==7)*pUSB=0;//备用
            if(IntI==8){//发送低速数据定义号,便于确定低速数据9的含义
               *pUSB=SendIndex;
               switch(SendIndex){
               case 0:SendValue=DOValue;break;
               case 1:SendValue=DAValue;break;
               case 2:SendValue=SPI_Data[0];break;
               case 3:SendValue=SPI_Data[1];break;
               case 4:SendValue=SPI_Data[2];break;
               case 5:SendValue=SPI_Data[3];break;
               case 6:SendValue=SPI_Data[4];break;
               default:SendValue=0;break;
               }//end of switch
            SendIndex++;if(SendIndex>=256)SendIndex=0;
          }
```

```
    if(IntI==9)*pUSB=SendValue;//发送低速数据
   }
   //后面的通道采集模拟量
   for(IntI=0;IntI<ADNumChn;IntI++){
       *pUSB=ADR[IntI];
   }
  }
 Timer1Count++;//系统计数器
if(Timer1Count==20000) Timer1Count=0;
 if((Timer1Count % 500)==0)ToggleL1();
 return ;
}
```

六、注意事项

1、不要带电插拔该板。

2、长期不使用时，建议从计算机中拔下该板，妥善保管。

3、本产品会根据用户需求不断推出新的分类功能，驱动程序和编程例程也会不断升级，一般情况下新软件会兼容旧软件，建议用户注意软件升级。

七、出库清单

1、RBH8281 板一块

2、专用 USB 接口电缆一根。

3、光盘一张（内含 demo 程序、驱动程序、校准程序、校准使用说明书、使用说明书等）

本公司拥有本卡的全部硬件和软件知识产权，为我公司独立设计和生产，能够为用户提供全面、周到的技术支持。